

[illegible]

3

Sy

MT

MT
MT

MT

MT

MT
MTMT
MTMT
MTMT
MT

MT
MT

MT
MT

MT

MT

MT
MT

MI
MT

MT
MT

MT
MT

MT
MT

MT
MT

MT

111

227

MT

MT
MT

MT

MT

M1
M1MT
MT

M1

M1

10

1

10

100

10

1

```
MM      MM      TTTTTTTTTT  HH      HH      GGGGGGGG  SSSSSSSS  IIIIII  NN      NN      HH      HH
MM      MM      TTTTTTTTTT  HH      HH      GGGGGGGG  SSSSSSSS  IIIIII  NN      NN      HH      HH
MMM     MMM     TT          HH      HH      GG          SS          II      NN      NN      HH      HH
MMM     MMM     TT          HH      HH      GG          SS          II      NN      NN      HH      HH
MM      MM      TT          HH      HH      GG          SS          II      NNNN   NN      HH      HH
MM      MM      TT          HH      HH      GG          SS          II      NNNN   NN      HH      HH
MM      MM      TT          HHHHHHHHHH  GG          SSSSSS  II      NN      NN      HHHHHHHHHH
MM      MM      TT          HHHHHHHHHH  GG          SSSSSS  II      NN      NN      HHHHHHHHHH
MM      MM      TT          HH      HH      GG      GGGGGG  SS          II      NN      NN      HH      HH
MM      MM      TT          HH      HH      GG      GGGGGG  SS          II      NN      NN      HH      HH
MM      MM      TT          HH      HH      GG          GG      SS          II      NN      NN      HH      HH
MM      MM      TT          HH      HH      GG      GGGGGG  SS          II      NN      NN      HH      HH
MM      MM      TT          HH      HH      GG          GG      SS          II      NN      NN      HH      HH
MM      MM      TT          HH      HH      GGGGGG  SSSSSSSS  IIIIII  NN      NN      HH      HH
MM      MM      TT          HH      HH      GGGGGG  SSSSSSSS  IIIIII  NN      NN      HH      HH

LL      IIIIII  SSSSSSSS
LL      IIIIII  SSSSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SSSSSS
LL      II      SSSSSS
LL      II      SS
LL      II      SS
LL      II      SS
LL      II      SS
LLLLLLLLLLLL IIIIII  SSSSSSSS
LLLLLLLLLLLL IIIIII  SSSSSSSS
```


(2)	50
(3)	70
(4)	131

HISTORY	: Detailed Current Edit History
DECLARATIONS	: Declarative Part of Module
MTH\$GSINH	- Standard G Floating DSINH


```
0000 1      .TITLE  MTH$GSINH      ; G Floating Hyperbolic Sine routine
0000 2      ; (GSINH)
0000 3      .IDENT /1-006/        ; File: MTH$GSINH.MAR  EDIT: JCW1006
0000 4      ;
0000 5      ;*****
0000 6      ;
0000 7      ; COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8      ; DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9      ; ALL RIGHTS RESERVED.
0000 10     ;
0000 11     ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12     ; ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13     ; INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14     ; COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15     ; OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16     ; TRANSFERRED.
0000 17     ;
0000 18     ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19     ; AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20     ; CORPORATION.
0000 21     ;
0000 22     ; DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23     ; SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24     ;
0000 25     ;*****
0000 26     ;
0000 27     ;
0000 28     ;
0000 29     ; FACILITY: MATH LIBRARY
0000 30     ; ++
0000 31     ; ABSTRACT:
0000 32     ;
0000 33     ; MTH$GSINH is a function which returns the G floating hyperbolic sine
0000 34     ; of its G floating point argument. The call is standard
0000 35     ; call-by-reference.
0000 36     ;
0000 37     ; --
0000 38     ;
0000 39     ; VERSION: 1
0000 40     ;
0000 41     ; HISTORY:
0000 42     ; AUTHOR:
0000 43     ; Steven B. Lionel, 26-Jan-79: Version 1
0000 44     ;
0000 45     ; MODIFIED BY:
0000 46     ;
0000 47     ;
0000 48     ;
```

```
0000 50      .SBTTL HISTORY ; Detailed Current Edit History
0000 51
0000 52
0000 53 ; ALGORITHMIC DIFFERENCES FROM FP-11/C ROUTINE: none
0000 54
0000 55 ; Edit History for Version 1 of MTH$GSINH
0000 56
0000 57 ; 1-001 - Adapted from MTH$DSINH version 1-001. SBL 26-Jan-79
0000 58 ; 1-002 - Use MTH$GEXP_R6. SBL 27-Sept-1979
0000 59 ; 1-003 - Eliminated second call to EXP for input values between 27*ln2
0000 60 ;           and 1023*ln2.
0000 61 ;           - Changed all final floating point divisions by 2 to interger
0000 62 ;           subtracts of 1 from the exponent field.
0000 63 ;           - Extended maximum range 1024*ln2.
0000 64 ;           - Changed logic for computing EXP(|x|-ln2) to reduce error.
0000 65 ;           - RNH 10-FEB-81
0000 66 ; 1-004 - Changed W^ to G^ in call to MTH$$SIGNAL RNH 09-Sept-1981
0000 67 ; 1-005 - Eliminated symbolic short literals. RNH 15-Oct-81
0000 68 ; 1-006 - Changed a MNEGD to a MNEGG. JCW 20-Jan-83
```



```
0000 70      .SBTTL  DECLARATIONS      ; Declarative Part of Module
0000 71
0000 72 :
0000 73 : INCLUDE FILES:
0000 74 :
0000 75 : EXTERNAL SYMBOLS:
0000 76      .DSABL  GBL
0000 77      .EXTRN  MTH$$SIGNAL
0000 78      .EXTRN  MTH$K_FLOOVEMAT
0000 79      .EXTRN  MTH$GEXP_R6
0000 80 :
0000 81 : EQUATED SYMBOLS:
0000 82
00000004 0000 83      value      = 4      ; value.rg.r
0000 84
0000 85 :
0000 86 : MACROS:      none
0000 87 :
0000 88
0000 89
0000 90 :
0000 91 : PSECT DECLARATIONS:
0000 92
00000000 0000 93      .PSECT  _MTH$CODE      PIC,SHR, LONG, EXE, NOWRT
0000 94      ; program section for math routines
0000 95 :
0000 96 : OWN STORAGE:  none
0000 97 :
0000 98 :
0000 99 : CONSTANTS:
0000 100 :
0000 101
0000 102 G_27_LOG_2:
20E28723 B7084052 0000 103      .QUAD      ^X20E28723B7084052      ; 27*ln2
0000 104 G_1023_LOG_2:
7B606E3A 28B740A6 0008 105      .QUAD      ^X7B606E3A28B740A6      ; 1023*ln2
0000 106 G_1024_LOG_2:
39EEFEFA 2E4240A6 0010 107      .QUAD      ^X39EEFEFA2E4240A6      ; 1024*ln2
0000 108 G_LOG_2_HI:
3C00FEFA 2E424006 0018 109      .QUAD      ^X3C00FEFA2E424006      ; (high 43 bits of ln2)+2**-43
0000 110 G_LOG_2_LO:
4C67361C 8654BD50 0020 111      .QUAD      ^X4C67361C8654BD50      ; ln2 - G_LOG_2_HI
0000 112
0000 113 GSINHTAB:
0000 114
EE8D 3E7A 0028 115      .WORD      ^0037172, ^0167215
CEC2 7466 002C 116      .WORD      ^0072146, ^0147302      ; DECIMAL: 0.2508223608819151D-07
1DE2 3EE7 0030 117      .WORD      ^0037347, ^0016742
AC9C 7B26 0034 118      .WORD      ^0075446, ^0126234      ; DECIMAL: 0.2755729803646086D-05
01A0 3F4A 0038 119      .WORD      ^0037512, ^0000640
4874 1A28 003C 120      .WORD      ^0015050, ^0044164      ; DECIMAL: 0.1984126984813681D-03
1111 3FA1 0040 121      .WORD      ^0037641, ^0010421
0ECD 1111 0044 122      .WORD      ^0010421, ^0007315      ; DECIMAL: 0.8333333333332327D-02
5555 3FE5 0048 123      .WORD      ^0037745, ^0052525
5556 5555 004C 124      .WORD      ^0052525, ^0052526      ; DECIMAL: 0.1666666666666667D+00
0000 0000 0050 125      .WORD      0,0
0000 0000 0054 126      .WORD      0,0      ; DECIMAL: 0.D0
```

MTH\$GSINH
1-006

F 6
: G Floating Hyperbolic Sine routine 16-SEP-1984 01:31:30 VAX/VMS Macro V04-00
DECLARATIONS ; Declarative Part of Modul 6-SEP-1984 11:24:16 [MTHRTL.SRC]MTH\$GSINH.MAR;1 Page 4
(3)

00000006 0058 127
0058 128 GSINHLEN = .- GSINHTAB/8
0058 129

MTH\$
1-006


```
0058 131 .SBTTL MTH$GSINH - Standard G Floating DSINH
0058 132
0058 133
0058 134 :++
0058 135 : FUNCTIONAL DESCRIPTION:
0058 136 :
0058 137 : GSINH - G floating point function
0058 138 :
0058 139 : GSINH(X) is computed as:
0058 140 :
0058 141 : If |X| < 2**(-27), GSINH(X) = X.
0058 142 : If 2**(-27) <= |X| < 0.25, GSINH(X) = Chebyshev Series.
0058 143 : If 0.25 <= |X| < 27*ln2, GSINH(X) = (GEXP(X) - GEXP(-X))/2.
0058 144 : If 27*ln2 <= |X| < 1023*ln2, GSINH(X) = sign(X)*GEXP(|X|)/2.
0058 145 : If 1023*ln2 <= |X| < 1024*ln2, then GSINH(X) = sign(X)*GEXP(|X|-ln2).
0058 146 : If 1024*ln2 <= |X|, then overflow.
0058 147 :
0058 148 : CALLING SEQUENCE:
0058 149 :
0058 150 : GSINH.wg.v = MTH$GSINH(x.rg.r)
0058 151 :
0058 152 : INPUT PARAMETERS:
0058 153 :
00000004 0058 154 : LONG = 4 ; define longword multiplier
00000004 0058 155 : x = 1 * LONG ; Contents of x is the argument
0058 156 :
0058 157 : IMPLICIT INPUTS: none
0058 158 :
0058 159 : OUTPUT PARAMETERS:
0058 160 :
0058 161 : VALUE: G floating hyperbolic sine of the argument
0058 162 :
0058 163 : IMPLICIT OUTPUTS: none
0058 164 :
0058 165 : COMPLETION CODES: none
0058 166 :
0058 167 : SIDE EFFECTS:
0058 168 :
0058 169 : Signal: MTH$ FLOOVEMAT if 1024*ln2 < |X| with reserved operand in R0/R1
0058 170 : (copied to the signal mechanism vector CHF$MCH_R0/R1 by LIB$SIGNAL).
0058 171 : Associated message is: "FLOATING OVERFLOW IN MATH LIBRARY". Result is reserved
0058 172 : operand -0.0 unless a user supplied (or any) error handler changes CHF$MCH_R0/R1
0058 173 :
0058 174 : NOTE: This procedure disables floating point underflow, enables integer
0058 175 : overflow.
0058 176 :
0058 177 : ---
0058 178 :
0058 179 :
40FC 0058 180 : .ENTRY MTH$GSINH, ^M<IV, R2, R3, R4, R5, R6, R7>
005A 181 : ; standard call-by-reference entry
005A 182 : ; disable DV (and FU), enable IV
005A 183 : MTH$FLAG_JACKET ; flag that this is a jacket procedure in
005A :
6D 00000000'GF 9E 005A : MOVAB G^MTH$$JACKET_HND, (FP)
0061 : ; set handler address to jacket
0061 : handler
```



```
0061 184
0061 185 ; case of an error in routine
0061 186 ; If an error, convert signal to user PC
0061 187 ; and resignal
56 04 BC 50FD 0061 188 MOVG @value(AP), R6 ; R6/R7 = |X| = @value(AP)
50 8000 8F 7D 0066 189 MOVQ R6, R0
3FF0 8F 50 B1 0069 190 BICW2 #^X8000, R0 ; R0/R1 = |X|
1E 18 006E 191 CMPW R0, #^X3FF0 ; compare |X| with 0.25
0073 192 BGEQ GEQ_TO_0.25 ; branch if |x| >= 0.25
0075 193
0075 194 ; |X| < 0.25
0075 195
0075 196
3E60 8F 50 B1 0075 197 CMPW R0, #^X3E60 ; compare |X| with 2**-27
04 18 007A 198 BGEQ GEQ_TO_2M27 ; branch if |x| >= 2**-27
007C 199
007C 200
007C 201 ; |X| < 2**-27
007C 202
007C 203
50 56 7D 007C 204 MOVQ R6, R0 ; R0/R1 = X
04 007F 205 RET ; return with result = argument
0080 206
0080 207 ; 2**-27 <= |X| < 0.25
0080 208
0080 209
0080 210
0080 211 GEQ_TO_2M27:
9E AF 50 50 44FD 0080 212 MULG2 R0, R0 ; Get ARG**2 for POLYG
05 50 55FD 0084 213 POLYG R0, #GSINHLN-1, GSINHTAB ; R0/R1 = SUM(Ci*X**i), with
008A 214 ; last coefficient zero
50 56 44FD 008A 215 MULG2 R6, R0 ; MULG2 by ARG, and then
50 56 40FD 008E 216 ADDG2 R6, R0 ; add in ARG with overhang.
04 0092 217 RET ; return with result in R0/R1
0093 218
0093 219
0093 220
0093 221 ; 0.25 <= |X|
0093 222
0093 223
0093 224 GEQ_TO_0.25:
FF6F CF 50 51FD 0093 225 CMPG R0, G_1023_LOG_2 ; compare |X| with 1023*ln2
3C 14 0099 226 BGTR GTR_THAN_1023_LOG_2 ; branch if |x| > 1023*ln2
009B 227
009B 228
009B 229 ; 0.25 <= |X| <= 1023*ln2
009B 230
009B 231
FF5F CF 50 51FD 009B 232 CMPG R0, G_27_LOG_2 ; Compare |X| to 27*ln2. If greater
20 14 00A1 233 BGTR ONLY_ONE_TERM ; only one call to GEXP is necessary.
00A3 234
00A3 235 ; 0.25 <= |X| < 27*ln2
00A3 236
00A3 237
50 56 7D 00A3 238 MOVQ R6, R0 ; R0/R1 = X
00000000 EF 16 00A6 239 JSB MTH$GEXP_R6 ; R0/R1 = GEXP(X)
```

```

      7E 50 7D 00AC 240      MOVQ   R0, -(SP)      ; push GEXP(X) on stack
      50 04 BC 52FD 241      MNEGG  @value(AP), R0  ; R0/R1 = -X
      00000000'EF 16 00B4 242      JSB   MTH$GEXP R6  ; R0/R1 = GEXP(-X)
      50 8E 50 43FD 243      SUBG3  R0, (SP)+, R0  ; R0/R1 = GEXP(X) - GEXP(-X)
      50 10 A2 00BF 244      SUBW   #*X0010, R0    ; R0/R1 = (GEXP(X)-GEXP(-X))/2
      04 00C2 245      RET                               ; return with result in R0/R1
      00C3 246
      00C3 247      ; 27*ln2 =< |X| < 1023*ln2
      00C3 248
      00C3 249
      00C3 250
      00C3 251      ONLY_ONE_TERM:
      00C3 252      JSB   MTH$GEXP R6      ; R0/R1 = GEXP(|X|)
      04 BC 53FD 00C9 253      TSTG   @value(AP)    ; Check sign of X
      04 14 00CD 254      BGTR   POSITIVE          ; If negative change sign of
      50 50 52FD 00CF 255      MNEGG  R0, R0      ; GEXP(|X|)
      00D3 256      POSITIVE:
      50 10 A2 00D3 257      SUBW   #*X0010, R0    ; R0/R1 = sign(X)*GEXP(|X|)/2
      04 00D6 258      RET
      00D7 259
      00D7 260      ; 1023*ln2 =< |X|
      00D7 261
      00D7 262
      00D7 263      GTR_THAN_1023_LOG_2:
      FF33 CF 50 51FD 00D7 264      CMPG   R0, G_1024_LOG_2  ; Compare |X| to 1024*ln2
      22 18 00DD 265      BGEQ   ERROR              ; if 1024*ln2 =<|X|, overflow occurs
      00DF 266
      00DF 267
      00DF 268
      00DF 269      ; 1023*ln2 < |X| < 1024*ln2
      00DF 270
      00DF 271
      00DF 272      SUBG2   G_LOG_2_HI, R0      ; R0/R1=|X|-G_LOG_2_HI
      50 FF34 CF 42FD 00DF 273      JSB   MTH$GEXP R6  ; R0/R1 = GEXP(|X|-G_LOG_2_HI)
      00000000'EF 16 00E5 274      MULG3  G_LOG_2_LO, R0, R2 ; R2/R3=GEXP(|X|-G_LOG_2_HI)*G_LOG_2_LO
      52 50 FF30 CF 45FD 00EB 275      SUBG2  R2, R0      ; R0/R1=GEXP(|X|-ln2)
      50 52 42FD 00F2 276      TSTG   @value(AP)    ; test the sign of X
      04 BC 53FD 00F6 277      BGEQ   10$          ; branch if X >= 0
      04 18 00FA 278      MNEGG  R0, R0      ; R0/R1 = sign(X) * GEXP(|X|-ln2)
      50 50 52FD 00FC 279      10$: RET              ; return with result in R0/R1
      04 0100 280
      0101 281
      0101 282
      0101 283
      0101 284      ; 1024*ln2 =< |X|, error
      0101 285
      0101 286
      7E 00'8F 9A 0101 287      ERROR: MOVZBL #MTH$K_FLOOVEMAT, -(SP) ; condition value
      50 01 0F 79 0105 288      ASHQ   #15, #T, R0    ; R0/R1 = result = reserved operand -0.0
      0109 289
      0109 290
      0109 291
      00000000'GF 01 FB 0109 292      CALLS  #1, G^MTH$$SIGNAL ; goes to signal mechanism vector
      0110 293
      0110 294      RET                               ; (CHF$MCH_R0/R1) so error handler
      0111 295
      0111 296      ; can modify the result.
      ; signal error and use real user's PC
      ; independent of CALL vs JSB
      ; return - R0/R1 restored from CHF$MCH_R0/
```


MTH\$GSINH
1-006

J 6
; G Floating Hyperbolic Sine routine
MTH\$GSINH - Standard G Floating DSINH

16-SEP-1984 01:31:30
6-SEP-1984 11:24:16

VAX/VMS Macro V04-00
[MTHRTL.SRC]MTHGSINH.MAR;1

Page 8
(4)

0111 297
0111 298 .END

MTH\$
VAX-
0 G
The
MAC

MTH\$GSINH
Symbol table

K 6
; G Floating Hyperbolic Sine routine

16-SEP-1984 01:31:30
6-SEP-1984 11:24:16

VAX/VMS Macro V04-00
[MTHRTL.SRC]MTH\$GSINH.MAR;1

Page 9
(4)

```
ERROR          00000101 R    01
GEQ_TO 0.25    00000093 R    01
GEQ_TO 2M27    00000080 R    01
GSINHLEN      = 00000006
GSINHTAB      00000028 R    01
GTR_THAN 1023_LOG_2 00000007 R    01
G_1023_LOG_2  00000008 R    01
G_1024_LOG_2  00000010 R    01
G_27_LOG_2    00000000 R    01
G_LOG_2_R1    00000018 R    01
G_LOG_2_LO    00000020 R    01
LONG          = 00000004
MTH$$JACKET_HND ***** X    01
MTH$$SIGNAL   ***** X    00
MTH$GEXP_R6   ***** X    00
MTH$GSINH     00000058 RG   01
MTH$K_FLOOVEMAT ***** X    00
ONLY ONE_TERM 000000C3 R    01
POSITIVE      000000D3 R    01
VALUE        = 00000004
```

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes												
ABS	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE		
MTH\$CODE	00000111 (273.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG		

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	32	00:00:00.07	00:00:00.36
Command processing	107	00:00:00.72	00:00:04.61
Pass 1	86	00:00:00.88	00:00:03.54
Symbol table sort	0	00:00:00.01	00:00:00.01
Pass 2	65	00:00:00.76	00:00:02.38
Symbol table output	4	00:00:00.04	00:00:00.40
Psect synopsis output	2	00:00:00.01	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	298	00:00:02.49	00:00:11.45

The working set limit was 900 pages.
4544 bytes (9 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 21 non-local and 1 local symbols.
358 source lines were read in Pass 1, producing 11 object records in Pass 2.
1 page of virtual memory was used to define 1 macro.

+-----+
! Macro library statistics !
+-----+

Macro library name

Macros defined

_ \$255\$DUA28:[SYSLIB]STARLET.MLB;2

0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:MTHGSINH/OBJ=OBJ\$:MTHGSINH MSRC\$:MTHJACKET/UPDATE=(ENH\$:MTHJACKET)+MSRC

0261 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY